UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/713,332	11/13/2003	Donald F. Hooper	Intel-006PUS	9877

7590 12/29/2006

Daly, Crowley & Mofford, LLP
c/o PortfolioIP
P.O. Box 52050
Minneapolis, MN 55402

EXAMINER	
CHEN, QING	
ART UNIT	**PAPER NUMBER**
2191	

SHORTENED STATUTORY PERIOD OF RESPONSE	MAIL DATE	DELIVERY MODE
3 MONTHS	12/29/2006	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

If NO period for reply is specified above, the maximum statutory period will apply and will expire 6 MONTHS from the mailing date of this communication.

Office Action Summary	Application No.	Applicant(s)	
	10/713,332	HOOPER ET AL.	
	Examiner	Art Unit	
	Qing Chen	2191	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *13 November 2003*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-27* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-27* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☒ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *19 April 2004* is/are: a)☐ accepted or b)☒ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date *20040429*.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

1. This is the initial Office action based on the application filed on November 13, 2003.

2. **Claims 1-27** are pending.

Drawings

3. The replacement drawings were received on April 19, 2004. The drawings are not acceptable because of non-compliance with 37 CFR § 1.121(d). Any changes to an application drawing must be in compliance with 37 CFR § 1.84 and must be submitted on a replacement sheet of drawings, which shall be an attachment to the amendment document and, in the top margin, labeled "Replacement Sheet."

4. The drawings are objected to as failing to comply with 37 CFR 1.84(p)(5) because they do not include the following reference sign(s) mentioned in the description:

- Reference numbers "62" and "64" on pages 9 and 10.

Corrected drawing sheets in compliance with 37 CFR 1.121(d) are required in reply to the Office action to avoid abandonment of the application.

The drawings are objected to as failing to comply with 37 CFR 1.84(p)(5) because they include the following reference character(s) not mentioned in the description:

- Reference number "200" in Figure 10.

Corrected drawing sheets in compliance with 37 CFR 1.121(d), or amendment to the specification to add the reference character(s) in the description in compliance with 37 CFR 1.121(b) are required in reply to the Office action to avoid abandonment of the application.

The drawings are objected to because:

- The word "Transfers" should be changed to singular form in Element 64a in Figure 2 as indicated in the specification *(see Page 9: 11-12)*.
- The word "Neighbors" should be changed to singular from in Element 74 in Figure 2 as indicated in the specification *(see Page 10: 19-20)*.
- The word "neighbors" should be changed to singular form in Elements "From previous neighbors ME" and "To Next Neighbor Regs (in next neighbor ME)" in Figure 2 as indicated in the specification *(see Page 11: 1-6)*.
- The reference numbers "76a" and "76b" should be changed to "75a" and "75b," respectively, on page 11.
- The letter "M" should be changed to "μ" in Elements 106 and 108 in Figure 4.
- The word "Code" should be changed to "μ Code" in Element 106 in Figure 5.
- Reference number "159" should be changed to "154" in Figure 6.

Corrected drawing sheets in compliance with 37 CFR 1.121(d) are required in reply to the Office action to avoid abandonment of the application.

Any amended replacement drawing sheet should include all of the figures appearing on the immediate prior version of the sheet, even if only one figure is being amended. The figure or

figure number of an amended drawing should not be labeled as "amended." If a drawing figure is to be canceled, the appropriate figure must be removed from the replacement sheet, and where necessary, the remaining figures must be renumbered and appropriate changes made to the brief description of the several views of the drawings for consistency. Additional replacement sheets may be necessary to show the renumbering of the remaining figures. Each drawing sheet submitted after the filing date of an application must be labeled in the top margin as either "Replacement Sheet" or "New Sheet" pursuant to 37 CFR 1.121(d). If the changes are not accepted by the Examiner, the Applicant will be notified and informed of any required corrective action in the next Office action. The objection to the drawings will not be held in abeyance.

Specification

5. The disclosure is objected to because of the following informalities:

- The specification contains the following typographical errors:
  - Reference number "74" should be changed to "66" on page 20, line 11.
  - Reference number "112" should be deleted on page 28, line 22, since reference number "112" is used to designate "Operand Navigation" as indicated in Figure 5.
  - Reference number "274" should be changed to "374" on page 47, line 7.
  - Reference number "280" should be changed to "380" on page 47, lines 8 and 11.
  - Reference number "282" should be changed to "382" on page 47, lines 8 and 16.
  - Reference number "374" should be changed to "370" on page 48, line 7.
  - Reference number "266" should be changed to "366" on page 48, line 13.

Appropriate correction is required.

Claim Objections

6. **Claims 9-17, 19, 25, and 27** are objected to because of the following informalities:

- **Claim 9** recites the limitation "using the attributes." Applicant is advised to change the limitation to read "looking up the attributes" for the purpose of providing it with proper explicit antecedent basis.
- **Claims 10-17** depend on Claim 9 and, therefore, suffer the same deficiency as Claim 9.
- **Claim 19** recites the limitation "the multiple execution threads." Applicant is advised to change the limitation to read "the multiple threads of execution" for the purpose of providing it with proper explicit antecedent basis.
- **Claim 25** contains a typographical error: Claim 25 should presumably depend on Claim 23, not Claim 22. In the interest of compact prosecution, the Examiner subsequently interprets this claim as depending on Claim 23 for the purpose of further examination.
- **Claim 27** contains a typographical error: the category of invention should be a device, not a system.

Appropriate correction is required.

Claim Rejections - 35 USC § 112

7. The following is a quotation of the second paragraph of 35 U.S.C. 112:

> The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

8. **Claims 14, 19, and 25** are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

Claim 14 recites the limitation "the physical register." There is insufficient antecedent basis for this limitation in the claim. In the interest of compact prosecution, the Examiner subsequently interprets this limitation as reading "the physical address" for the purpose of further examination.

Claim 19 recites the limitation "the microcode." There is insufficient antecedent basis for this limitation in the claim. In the interest of compact prosecution, the Examiner subsequently interprets this limitation as reading "the instructions" for the purpose of further examination. Consequently, the word "is" after "the instructions" should be changed to "are," since "instructions" is in plural form.

Claim 25 recites the limitation "the first window." There is insufficient antecedent basis for this limitation in the claim. In the interest of compact prosecution, the Examiner subsequently interprets this limitation as reading "the window" for the purpose of further examination.

Claim Rejections - 35 USC § 102

9. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

10. **Claims 1-9 and 18-22** are rejected under 35 U.S.C. 102(b) as being anticipated by **Wilmot, II (US 5,974,538)**.

As per **Claim 1**, Wilmot, II discloses:

- receiving a user selection of a first instruction from a list of instructions that executed during a processor simulation *(see Figure 2; Figure 3: 302; Column 6: 16-26, "... when instruction 201 is executed in operand flow mapping mode ..."; Column 31: 59-61, "The microarchitecture of the present invention can be simulated by software programs for any instruction set architecture.")*; and

- tracing an operand in the first instruction directly to a use of the operand in a second instruction in the list of instructions by following operand dependencies between such first and second instructions *(see Figure 2; Figure 3: 304; Column 6: 16-26, "One or more unrelated instructions such as 202 may be executed followed by execution of an instruction 203 using the operand 206 as input.")*.

As per **Claim 2**, the rejection of **Claim 1** is incorporated; and Wilmot, II further discloses:

- wherein tracing determines that the second instruction set the value of the operand as used in the first instruction as a source operand *(see Figure 3: 302 and 304)*.

As per **Claim 3**, the rejection of **Claim 1** is incorporated; and Wilmot, II further discloses:

- wherein tracing determines that a next use of the operand, after that of the first instruction as a destination operand, occurs in the second instruction *(see Column 7: 48-52, "When operands are modified within iterations of a loop, those new values might be used by other instructions in the same iteration or in instructions in subsequent iterations of the same loop or in instructions following after termination of the loop or in any combination of these flows.")*.

As per **Claim 4**, the rejection of **Claim 1** is incorporated; and Wilmot, II further discloses:

- determining attributes of the first instruction *(see Column 6: 16-19, "... when instruction 201 is executed in operand flow mapping mode, its operand 206 is annotated with the address of instruction 201 as well as having the stored value updated to the value of '6' in that example.")*; and

- using the attributes of the first instruction to find the second instruction *(see Figure 2; Column 6: 22-26, "As the operand (sic) 201 is accessed in operand mapping mode, the address*

of the last instruction to alter it 205 is also retrieved and used to establish the operand flow. The operand flow 204 is then associated with sending instruction 201.").

As per **Claim 5**, the rejection of **Claim 4** is incorporated; and Wilmot, II further discloses:

- receiving a selected cycle corresponding to the first instruction *(see Column 7: 42-47 "... trigger immediate execution of instruction 505 if any execution pipelines are available or make it eligible for execution in an upcoming clock cycle.").*

As per **Claim 6**, the rejection of **Claim 5** is incorporated; and Wilmot, II further discloses:

- determining a program counter value associated with the selected cycle *(see Column 6: 1-5, "... it may be desirable to treat the program counter (PC) as a register type operand which is forwarded from an instruction to the instruction that should execute next at least for some sequences of instructions.").*

As per **Claim 7**, the rejection of **Claim 6** is incorporated; and Wilmot, II further discloses:

- using the program counter value to look up the attributes in an instruction operand map that provides attributes of each instruction, including instruction type and type of registers used by such instruction type for operands *(see Column 5: 40-49, "All of these operand flows can be mapped as proceeding from one (operand-altering or sending) instruction to another*

(operand-using or target) instruction and each of the flows of an operand can be associated with its sending instruction. Operands that can be mapped include registers, control registers, vectors, storage locations, condition codes, stack locations and the program counter. A computer designer might design a computer to map and forward any one or several of the operand types and not others.").

As per **Claim 8**, the rejection of **Claim 7** is incorporated; and Wilmot, II further discloses:

- wherein the instructions are instructions of a microcode and the instruction operand map is generated at microcode build time *(see Abstract, "Values for registers, condition codes, stack locations and memory storage locations are routed directly from the program instructions or microcode that alter them to the instructions that use those operands."; Column 31: 2-9, "Such compiled mappings provide a performance advantage ...").*

As per **Claim 9**, the rejection of **Claim 7** is incorporated; and Wilmot, II further discloses:

- determining for each register type a physical address *(see Column 3: 11-16, "The present invention maps the flow of operands among instructions in a first Operand Flow Mapping Mode of operation where each operand (such as a register, condition code, vector, stack or memory location) is annotated with an identifier (e.g., address) of the last instruction to modify that operand.").*

As per **Claim 18**, the rejection of **Claim 1** is incorporated; and Wilmot, II further discloses:

- wherein the instructions are intended for execution on at least one microengine of the processor simulated by the processor simulation *(see Column 4: 63-66, "Instructions from the pool store 1B7 connect over bus 1B8 to the one or more execution units such as execution units 1B9, 1B10 and 1B11.").*

As per **Claim 19**, the rejection of **Claim 18** is incorporated; and Wilmot, II further discloses:

- wherein the microengine is configured to support multiple threads of execution and the instructions are intended for execution by at least one of the multiple threads of execution *(see Figure 13; Column 24: 28-34, "From among a number of instances of a forwarded operand an instruction can be use the one operand for a particular operand slot that matches the current thread number. This operation follows continued parallelism where several threads are making way through the same areas of program code because the processor will have associated a saved set of operands with a thread and an interrupt address.").*

As per **Claim 20**, Wilmot, II discloses:

- a storage medium having stored thereon instructions *(see Figure 1B: 1B7)* that when executed by a machine result in the following:
  - receiving a user selection of a first instruction from a list of instructions that executed during a processor simulation *(see Figure 2; Figure 3: 302; Column 6: 16-26, "...*

when instruction 201 is executed in operand flow mapping mode ... "; Column 31: 59-61, "The microarchitecture of the present invention can be simulated by software programs for any instruction set architecture."); and

- tracing an operand used in the first instruction directly to a use of the operand in a second instruction in the list of instructions by following operand dependencies between such first and second instructions *(see Figure 2; Figure 3: 304; Column 6: 16-26, "One or more unrelated instructions such as 202 may be executed followed by execution of an instruction 203 using the operand 206 as input.")*.

As per **Claim 21**, the rejection of **Claim 20** is incorporated; and Wilmot, II further discloses:

- wherein tracing determines that the second instruction set the value of the operand as used in the first instruction *(see Figure 3: 302 and 304)*.

As per **Claim 22**, the rejection of **Claim 20** is incorporated; and Wilmot, II further discloses:

- wherein tracing determines that a next use of the operand after that of the first instruction occurs in the second instruction *(see Column 7: 48-52, "When operands are modified within iterations of a loop, those new values might be used by other instructions in the same iteration or in instructions in subsequent iterations of the same loop or in instructions following after termination of the loop or in any combination of these flows.")*.

11. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

12. **Claims 26 and 27** are rejected under 35 U.S.C. 102(e) as being anticipated by **Muratori et al. (US 6,611,276)**.

As per **Claim 26**, Muratori et al. disclose:

- at least one line card for forwarding networking data to ports of a switching fabric *(see Figure 1: 16; Column 3: 55-64, "Processor 10 interfaces to network devices 30 on Fbus 32 through Fbus interface unit 16. Fbus 32 is a 64-bit wide FIFO (First-In First-Out) bus. Fbus interface unit 16 contains transmit and receive buffers and a FIFO bus interface to network devices 30.")*;

- the at least one line card comprising a network processor comprising multi-threaded microengines each configured for execution with a microcode *(see Figure 1: 10; Column 3: 1-5, "Processor 10 includes core 12, microengines 14, Fbus interface unit 16 ..." and 13-14, "Microengines 14 support up to four threads per engine, which are executed in parallel to perform various tasks.")*; and

- wherein the microcode comprises a microcode developed using a debugger tool that allowed tracing of operands in code lines of the microcode once executed by a simulator

simulating operation of the network processor *(see Figures 3 and 6; Column 4: 16-28, "... computer instructions 48 generate a GUI (for display on display screen 50) which allows a programmer to view results of the simulations and to debug computer code in threads 40.").*

As per **Claim 27**, the rejection of **Claim 26** is incorporated; and Muratori et al. further disclose:

- wherein the operands are associated with registers in the microengines, and the registers include general purpose registers and I/O transfer registers *(see Column 3: 6-10, "Core 12 is a central controller that manages the resources of processor 10, loads microcode threads into microengines 14 ..." It is inherent that registers are components inside a processor for storing commonly used values. It is also inherent that general purpose registers (GPRs) and I/O transfer registers are classes of registers available to a processor.).*

Claim Rejections - 35 USC § 103

13. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

14. **Claims 10-14** are rejected under 35 U.S.C. 103(a) as being unpatentable over **Wilmot, II (US 5,974,538)** in view of **Celtruda et al. (US 5,148,538)**.

As per **Claim 10**, the rejection of **Claim 9** is incorporated; however, Wilmot, II does not disclose:

- wherein determining the physical address comprises determining whether each register type is a non-I/O register or an I/O register.

Celtruda et al. disclose:

- wherein determining the physical address comprises determining whether each register type is a non-I/O register or an I/O register *(see Column 6: 34-38, "The operand of the instruction is a series of bits that is divided into three sections and two of the sections specify a register within the computer which holds information to be used in generating a desired address of data within cache memory.")*.

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Celtruda et al. into the teaching of Wilmot, II to include wherein determining the physical address comprises determining whether each register type is a non-I/O register or an I/O register. The modification would be obvious because one of ordinary skill in the art would be motivated to efficiently retrieve the desired physical address.

As per **Claim 11**, the rejection of **Claim 10** is incorporated; however, Wilmot, II does not disclose:

- wherein determining the physical address comprises determining whether each non-I/O register is accessed using an index register.

Celtruda et al. disclose:

- wherein determining the physical address comprises determining whether each non-I/O register is accessed using an index register *(see Column 6: 38-43, "The third section is itself used in the generation of the desired address. The high order bits specify an index register and the middle group of bits specify a base register. The index and base registers are part of the general purpose registers (GPRs) 320 of the computer.")*.

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Celtruda et al. into the teaching of Wilmot, II to include wherein determining the physical address comprises determining whether each non-I/O register is accessed using an index register. The modification would be obvious because one of ordinary skill in the art would be motivated to efficiently retrieve the desired physical address.

As per **Claim 12**, the rejection of **Claim 11** is incorporated; however, Wilmot, II does not disclose:

- wherein the instruction operand map is used to provide the physical address for each non-I/O register that is not accessed using an index register.

Celtruda et al. disclose:

- wherein the instruction operand map is used to provide the physical address for each non-I/O register that is not accessed using an index register *(see Column 6: 43-51, "The lower order group are called displacement bits 16. The contents of the index register and the base register are added, by an adder 310 to the displacement bits 16 to produce a virtual address. The virtual address is placed in the virtual address register (VAR) 40 and is used to index a table,*

called the translation look aside buffer (TLB) 420, which contains the desired or real address of the data in cache memory.").

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Celtruda et al. into the teaching of Wilmot, II to include wherein the instruction operand map is used to provide the physical address for each non-I/O register that is not accessed using an index register. The modification would be obvious because one of ordinary skill in the art would be motivated to efficiently retrieve the desired physical address.

As per **Claim 13**, the rejection of **Claim 11** is incorporated; however, Wilmot, II does not disclose:

- wherein the physical address for each non-I/O register that is determined to be accessed using an index register is determined by obtaining a historical value of the index register at the selected cycle from a register history that records historical values of registers for each register type as such values change during simulation.

Celtruda et al. disclose:

- wherein the physical address for each non-I/O register that is determined to be accessed using an index register is determined by obtaining a historical value of the index register at the selected cycle from a register history that records historical values of registers for each register type as such values change during simulation *(see Column 4: 58-67, "The history table is a list of recently used current addresses and the corresponding real addresses that were generated by the verification part of the cache memory access system for those current*

addresses. A part of the current address is compared to a corresponding part of each address in the list of current addresses of the history table. When the part of the current address matches a part of a current address in the history table, the corresponding recently used real address becomes the predicted real address.").

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Celtruda et al. into the teaching of Wilmot, II to include wherein the physical address for each non-I/O register that is determined to be accessed using an index register is determined by obtaining a historical value of the index register at the selected cycle from a register history that records historical values of registers for each register type as such values change during simulation. The modification would be obvious because one of ordinary skill in the art would be motivated to efficiently retrieve the desired physical address.

As per **Claim 14**, the rejection of **Claim 12** is incorporated; however, Wilmot, II does not disclose:

- wherein the physical address for any register determined to be an I/O register is obtained for the selected cycle from a memory reference history that records physical addresses and reference counts for each of the I/O registers that is used in a memory reference during simulation.

Celtruda et al. disclose:

- wherein the physical address for any register determined to be an I/O register is obtained for the selected cycle from a memory reference history that records physical addresses

and reference counts for each of the I/O registers that is used in a memory reference during simulation *(see Column 5: 45-54, "The TLAT is, like the history table, a list of recently used current addresses and their corresponding real addresses generated by the TLB. However, it is distinct from the history table in that it is smaller than the history table and is updated on each instruction decode machine cycle rather than on each error in comparison between the predicted real address and the real address.").*

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Celtruda et al. into the teaching of Wilmot, II to include wherein the physical address for any register determined to be an I/O register is obtained for the selected cycle from a memory reference history that records physical addresses and reference counts for each of the I/O registers that is used in a memory reference during simulation. The modification would be obvious because one of ordinary skill in the art would be motivated to efficiently retrieve the desired physical address.

15. **Claims 15-17** are rejected under 35 U.S.C. 103(a) as being unpatentable over **Wilmot, II (US 5,974,538)** in view of **Swoboda et al. (US 5,564,028)**.

As per **Claim 15**, the rejection of **Claim 9** is incorporated; however, Wilmot, II does not disclose:

- wherein determining the program counter value comprises looking up the program counter value in a program counter history that records state change events, which are detected

during simulation, with associated program counter values for each cycle in which such state change events occurred.

Swoboda et al. disclose:

- wherein determining the program counter value comprises looking up the program counter value in a program counter history that records state change events, which are detected during simulation, with associated program counter values for each cycle in which such state change events occurred *(see Column 5: 13-28, "The IPA 39 and IPE 41 are referred to as history registers because they contain a history of the last two program counter addresses." and "Thus, at any given time, the program counter 21 holds the address of the instruction being fetched, the IPA39 holds the address of the instruction in the IRA 23, and the IPE 41 holds the address of the instruction in the IRE 25, whereby the instructions (by virtue of IRA 23 and IRE 25) advance through an "instruction pipeline" and their addresses (by virtue of IPA 39 and IPE 41) advance through an "address pipeline".").*

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Swoboda et al. into the teaching of Wilmot, II to include wherein determining the program counter value comprises looking up the program counter value in a program counter history that records state change events, which are detected during simulation, with associated program counter values for each cycle in which such state change events occurred. The modification would be obvious because one of ordinary skill in the art would be motivated to trace only those instructions which are actually executed during the trace period *(see Swoboda et al. – Column 1: 30-31).*

As per **Claim 16**, the rejection of **Claim 15** is incorporated; however, Wilmot, II does not disclose:

- using the physical address for each register used in the first instruction to traverse the program counter history, instruction by instruction, to find a matching physical address in the second instruction.

Swoboda et al. disclose:

- using the physical address for each register used in the first instruction to traverse the program counter history, instruction by instruction, to find a matching physical address in the second instruction *(see Figure 7; Column 8: 65-67 through Column 9: 1-5, "Fetches from the addresses indicated by the program counter are not actually performed during clock cycles 4 and 5, but the instructions at A and B are executed and the program counter history is recorded by advancing the address pipeline during clock cycles 4 and 5.").*

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Swoboda et al. into the teaching of Wilmot, II to include using the physical address for each register used in the first instruction to traverse the program counter history, instruction by instruction, to find a matching physical address in the second instruction. The modification would be obvious because one of ordinary skill in the art would be motivated to trace only those instructions which are actually executed during the trace period *(see Swoboda et al. – Column 1: 30-31).*

As per **Claim 17**, the rejection of **Claim 16** is incorporated; and Wilmot, II further discloses:

- wherein the microcode is intended for execution on one or more microengines in a processor simulated by the processor simulation *(see Column 4: 63-66, "Instructions from the pool store 1B7 connect over bus 1B8 to the one or more execution units such as execution units 1B9, 1B10 and 1B11.").*

However, Wilmot, II does not disclose:

- and wherein the program counter history of more than one of the microengines is traversed.

Swoboda et al. disclose:

- and wherein the program counter history of more than one of the microengines is traversed *(see Figure 7; Column 8: 65-67 through Column 9: 1-5, "Fetches from the addresses indicated by the program counter are not actually performed during clock cycles 4 and 5, but the instructions at A and B are executed and the program counter history is recorded by advancing the address pipeline during clock cycles 4 and 5.").*

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Swoboda et al. into the teaching of Wilmot, II to include and wherein the program counter history of more than one of the microengines is traversed. The modification would be obvious because one of ordinary skill in the art would be motivated to trace only those instructions which are actually executed during the trace period *(see Swoboda et al. – Column 1: 30-31).*

16. **Claims 23-25** are rejected under 35 U.S.C. 103(a) as being unpatentable over **Muratori et al. (US 6,611,276)** in view of **Lindsey (US 5,896,536)**.

As per **Claim 23**, Muratori et al. disclose:

- a window showing a view of microcode instructions that executed on a processor simulator during a simulation and for which a simulation history has been collected by the processor simulator *(see Figures 3 and 6; Column 4: 16-28, "... computer instructions 48 generate a GUI (for display on display screen 50) which allows a programmer to view results of the simulations and to debug computer code in threads 40." and "GUI 52 shows the operational history of computer code in threads 40 identified in display area 54." and 51-52, "The states of execution are obtained by computer instructions 48 from routine(s) simulating processor 10.")*; and

- the view being usable to provide a tracing option in a menu presented to a user for one of the instructions as an instruction of interest *(see Column 8: 20-25, "Pointer 192 in thread window 190 identifies the code that is executing at the processor cycle identified in field 100. Pointer 192 is movable in synchronism with pointer 98, and moves relative to computer code 196 in window 190 to indicate which portion of computer code 196 is executing at the time identified in field 100.")*.

However, Muratori et al. do not disclose:

- the tracing option being usable to trace any variable used by the instruction of interest in the simulation history directly to a second instruction in which a most recent change to or next use of such variable occurred.

Lindsey discloses:

- the tracing option being usable to trace any variable used by the instruction of interest in the simulation history directly to a second instruction in which a most recent change to or next use of such variable occurred *(see Figure 5; Column 6: 47-54, "FIG. 5 illustrates a window 80 which may be displayed in a user interface of a software tool which provides the developer with options available for the tracing being set relative to the selected instances variables. For example, the developer can specify which access of the instance variable (i.e., the first, tenth, eightieth) during execution of the object oriented program will cause the tracing function to be turned on by setting a number in box 82.").*

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Lindsey into the teaching of Muratori et al. to include the tracing option being usable to trace any variable used by the instruction of interest in the simulation history directly to a second instruction in which a most recent change to or next use of such variable occurred. The modification would be obvious because one of ordinary skill in the art would be motivated to gather trace information specific to the designated data components *(see Lindsey – Column 2: 39-43).*

As per **Claim 24**, the rejection of **Claim 23** is incorporated; however, Muratori et al. do not disclose:

- wherein selection of the tracing option by the user causes a submenu of options available for the instruction of interest to be provided to the user, each of the options of the submenu corresponding to one of the variables used by the instruction of interest.

Lindsey discloses:

- wherein selection of the tracing option by the user causes a submenu of options available for the instruction of interest to be provided to the user, each of the options of the submenu corresponding to one of the variables used by the instruction of interest *(see Figure 4; Column 6: 40-46, "A scrollable list 72 of data elements or instance variables is provided in the window 70. A number of selectable boxes 74, one logically associated with each data item in the list 72, is also provided in the window 70. The developer sets a tracepoint relative to a given instance variable from the list 72 by selecting the box 74 associated with the desired instance variable.")*.

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Lindsey into the teaching of Muratori et al. to include wherein selection of the tracing option by the user causes a submenu of options available for the instruction of interest to be provided to the user, each of the options of the submenu corresponding to one of the variables used by the instruction of interest. The modification would be obvious because one of ordinary skill in the art would be motivated to gather trace information specific to the designated data components *(see Lindsey – Column 2: 39-43)*.

As per **Claim 25**, the rejection of **Claim 23** is incorporated; and Muratori et al. further disclose:

- a second window in which a cycle of interest corresponding to the instruction of interest is indicated *(see Figures 3 and 5; Column 5: 33-47, "Information 96 identifies the cycle at point 94 of thread 58c ("Cycle=4694") ...")*;

- wherein the indication of the cycle of interest is modified to indicate a new cycle of interest corresponding to the second instruction *(see Column 5: 48-57, "GUI 52 also includes a pointer that identifies a particular processor cycle in one-cycle increments. The pointer is not shown in FIG. 3 or 5 (scrolling right using scroll bar 88 would reveal the pointer at cycle 5263) ..." and "Arrows 102a and 102b move this pointer left and right, respectively ...")*; and

- wherein the window is modified to reflect the new cycle of interest *(see Column 5: 41-44, "Pointing (without clicking) a mouse (or other input device) to a state indicator causes information to be displayed to the user. For example, as shown in FIG. 5, pointing to point 92 on state indicator 94 displays information 96.")*.

Conclusion

17. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

A. **Iwasaki et al. (US 4,879,646)** disclose an information processing system for debugging the software to be developed by the processor.

B. **Fogg, Jr. et al. (US 4,951,195)** disclose a system and method of simulating the first processor for running the applications on a second processing system having a second dissimilar processor.

C. **Luke (US 5,168,554)** discloses transforming discrete functional events recorded in trace data into time process diagrams depicting parallel activity duration, parallel process summary variables, and concurrency variables for evaluative purposes.

D. **Levine et al. (US 5,446,876)** disclose a tracing mechanism, implemented in system hardware, that allows for trace and profile characteristics of instructions executed by a central processing unit (CPU) to be generated for use by a system designer.

E. **Lesartre et al. (US 5,761,474)** disclose a system and method for tracking operand dependencies among instructions in a processor that executes instructions out of order.

F. **Sumi et al. (US 5,881,288)** disclose a program development system which is made up of a debugging apparatus which, during software development, can check that execution code generated by a program conversion apparatus operates as intended and which can quickly discover any errors present in such code, and a debugging information generation apparatus for generating debugging information for the debugging apparatus.

G. **Diec et al. (US 6,083,281)** disclose a process and apparatus for tracing entities operating in a computer system for the purpose of software debugging.

H. **Grochowski (US 6,088,790)** discloses a table for tracking operand locations in a processor pipeline.

I. **Edwards (US 6,463,553)** discloses methods and apparatus for carrying out debugging operations on microcomputers.

J. **DaSilva et al. (US 6,493,868)** disclose an integrated code development tool, comprising of an editor, a project management and build system, a debugger, a profiler, and a graphical data visualization system.

K. **Alverson et al. (US 6,848,097)** disclose a system for debugging targets using various techniques, some of which are particularly useful in a multithread environment.

L. **Dzoba et al. (US 7,055,136)** disclose software and hardware testing tools for debugging systems that include one or more processing units or microprocessors.

Any inquiry concerning this communication or earlier communications from the Examiner should be directed to Qing Chen whose telephone number is 571-270-1071. The Examiner can normally be reached on Monday through Thursday from 7:30 AM to 4:00 PM. The Examiner can also be reached on alternate Fridays.

If attempts to reach the Examiner by telephone are unsuccessful, the Examiner's supervisor, Wei Zhen, can be reached on 571-272-3708. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Any inquiry of a general nature or relating to the status of this application or proceeding should be directed to the TC 2100 Group receptionist whose telephone number is 571-272-2100.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR

system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

WEI ZHEN
SUPERVISORY PATENT EXAMINER

QC / QC
December 18, 2006